Solaris FCA Utilities

Version 1.06

User Manual

P007005-01A Rev. A

Emulex Connects™
Servers, Storage and People

Copyright © 2003-2011 Emulex. All rights reserved worldwide. No part of this document may be reproduced by any means or translated to any electronic medium without the prior written consent of Emulex.

Information furnished by Emulex is believed to be accurate and reliable. However, no responsibility is assumed by Emulex for its use; or for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright or related rights of Emulex.

Emulex, the Emulex logo, AutoPilot Installer, AutoPilot Manager, BlockGuard, Connectivity Continuum, Convergenomics, Emulex Connect, Emulex Secure, EZPilot, FibreSpy, OneCommand Manager, InSpeed, LightPulse, MultiPulse, OneCommand, OneConnect, One Network. One Company., SBOD, SLI, and VEngine are trademarks of Emulex. All other brand or product names referenced herein are trademarks or registered trademarks of their respective companies or organizations.

Emulex provides this manual "as is" without any warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability or fitness for a particular purpose. Emulex may make improvements and changes to the product described in this manual at any time and without any notice. Emulex assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties that may result. Periodic changes are made to information contained herein; although these changes will be incorporated into new editions of this manual, Emulex disclaims any undertaking to give notice of such changes.

Emulex, 3333 Susan Street
Costa Mesa, CA 92626

Introduction 1
Installing the Utilities 1
Compatibility 1
Installing or Updating the Utilities Using the emlxu_install Script 1
Removing the Utilities Using the emlxu_remove Script 3
Installing the Utilities Package Manually 4
Removing the Utilities Package Manually 4
Updating the Utilities Package Manually 5
Using the emlxadm Utility 6
Modes of Operation (emlxadm) 6
Interactive Mode (emlxadm) 6
CLI Mode (emlxadm) 8
Device Path Option 8
Instance option 10
Using Help 10
Command Descriptions (emlxadm) 11
Command Usage 14
boot_code [enable or disable] 14
dev_login <wwpn> 14
dev_logout <wwpn> 14
dev_remove <wwpn> 15
diag <test [parameters]> or diag code <cmd_code (hex)> 15
download_boot <filename> 15
download_fcode <filename> 16
download_fw <filename> 16
force_dump 16
get_boot_rev 17
get_dev_list 17
get_dump_size 18
get_dump [-h] 18
get_fcode_rev 18
get_fw_rev 18
get_hba_attrs 18
get_host_attrs 19
get_host_params 20
get_inst 21
get_linkinfo 21
get_logi_params <wwpn> 21
get_num_devs 22
get_path <index> 22
get_phy_attrs 22
get_port_attrs <index>, <wwn> or all 22
get_rnid [wwpn] 24
get_state <wwpn> 25
get_sym_pname 25
get_sym_nname 25
get_topology 25
get_vpd 25
h 26
hba 26
link_status <d_id> 27
msgbuf all or <number> [-i interval] 27
ns 27

parm_get_num 28
parm_get <label> 29
parm_get_list 29
parm_set <label> <value> 32
p 32
q 32
reset_hard 33
reset_link <wwpn or zero for local link> 33
set_sym_nname <"string"> 33
set_sym_pname <"string"> 33
Using the emlxdrv Utility 34
Modes of Operation (emlxdrv) 34
Interactive Mode (emlxdrv) 34
CLI Mode (emlxdrv) 36
Command Descriptions (emlxdrv) 37
Command Usage 38
clear [driver] all 38
clear [driver] [alias] 38
clear [driver] emulex 39
clear [driver] oracle 40
refresh 40
set emlxs [driver_name] [alias] 41
set [driver] all 41
set [driver] oracle 42
set [driver name] [emulex] 42

Introduction

This document provides the information needed to use the Emulex® emlxadm and emlxdrv utility programs. Emulex emlxadm and emlxdrv utility programs support the following drivers:

- emlxs - Solaris inbox Fibre Channel/Fibre Channel over Ethernet (FC/FCoE) driver
- elxfc - Emulex distributed FC/FCoE Solaris driver (does not support Oracle-branded devices)
- lpfc FC - Legacy Emulex distributed FC SD driver (does not support Oracle-branded devices)
- oce - Solaris inbox Network Interface Card (NIC) driver
- elxnic - Emulex distributed Solaris NIC driver

For system administrators, this document includes information about installing and uninstalling these utilities.

System administrators should be familiar with Solaris and the Oracle StorEdge SAN Foundation Software (SFS) and should have access to standard system documentation. Anyone working with this product should also be familiar with the Emulex OneConnect UCNA, as well as Fibre Channel and Ethernet networking.

Installing the Utilities

The Emulex utility programs are provided on the Emulex website. Download the utilities from the Emulex website after selecting the appropriate Solaris SFS operating system version.

Compatibility

The Emulex drivers for Solaris and the FCA utilities support the following operating systems:

- Solaris 10 for SPARC
- Solaris 10 for x64 and x86
- Solaris 11 Express for SPARC
- Solaris 11 Express for x64 and x86

To determine which adapters are supported by the Solaris SFS FCA (emlxs, elxfc) drivers, Solaris Ethernet NIC (oce, elxnic) drivers and the Emulex Fibre Channel adapter utilities (FCA utilities), refer to the Emulex website.

Note: The OneCommand Manager application is supported only for LightPulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Installing or Updating the Utilities Using the emlxu_install Script

Although it is possible to install emlxu onto one or more clients from a server, that procedure is not covered in this document; refer to the Solaris documentation.

Before installing the Emulex emlxu utilities package, you must completely install the Solaris 10 Update 8 (Solaris 10 10/09) with Oracle-recommended patches. Make sure the emlxs driver is v2.50o (or later) and the oce driver is v1.10c (or later).

If an earlier version of the emlxu utilities package is already installed on the system, the emlxu_install script removes the old version before installing the new version.

To install the utilities kit using the emlxu_install script:

1. Log in as root, or su to root.
2. Copy the utilities kit from your distribution medium into a directory, referred to here as <directory>. The utilities kit is a .tar file named something similar to emlxu_kit-<version>-sparc.tar.
3. Change to the directory where you put the kit tar file by typing:

   ```
   cd <directory>
   ```

4. Extract the emlxu_install script from the tar file by typing:

   ```
   tar xf emlxu_kit-<version>-sparc.tar emlxu_install
   ```

5. Install the kit by typing:

   ```
   emlxu_install
   ```

   The script removes any earlier version of the emlxu utilities package. (If an earlier package is not found, this fact is indicated; skip to step 9.) The following text is displayed:

   ```
   <Removing old EMLXemlxu package>
   ```

6. If an old package is installed, you are prompted to remove it:

   ```
   Do you want to remove this package? [y,n,?,q]
   ```

   Enter **<y>**. The following message is displayed:

   ```
   Removal of <EMLXemlxu> was successful.
   ```

   The script expands the utilities kit .tar file and begins installing the new package. The following message is displayed:

   ```
   <Expanding emlxu_kit-<version>-sparc.tar>
   <Adding new package>
   ```

7. The script installs the emlxu utilities package. The package is prepared for installation and you are prompted for confirmation by the following message:

   ```
   Do you want to continue with the installation of <EMLXemlxu> [y,n,?]:
   ```

   Enter **<y>**. The installation progress is indicated.

8. Examine the output for any errors or warnings. If the installation is successful, the following message is displayed near the end of the process:

   ```
   Installation of <EMLXemlxu> was successful.
   ```

   The script performs some cleanup and the following messages are displayed:

   ```
   <Cleaning directory>
   <emlxu_install complete>
   <Execute "emlxu_remove" when ready to uninstall>
   ```

   The script leaves a copy of the emlxu_remove script in the working directory with the original utilities kit tar file. You can remove this script, or leave it in the directory and use it to uninstall the emlxu utilities from your system in the future. See “Removing the Utilities Using the emlxu_remove Script” on page 3 for more details.

The emlxu utilities installation is complete. The utility package's programs are located in the /opt/EMLXemlxu/bin directory.

You do not have to reboot the system to run a utility program, but you must either enter the program’s full path name, or add the package's bin directory (/opt/EMLXemlxu/bin) to the system environment’s search path. To use the man pages provided by the package, you must also add the package's man directory (opt/EMLXemlxu/man) to the system environment's man path.

For further information on installing and removing packages, refer the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Removing the Utilities Using the emlxu_remove Script

You can uninstall the utilities kit using the emlxu_remove script. If you do not have the emlxu_remove script and you do not have the original emlxu utilities kit tar file, you must uninstall the emlxu package manually using the instructions in “Removing the Utilities Package Manually” on page 4. If you are updating the emlxu utilities to a newer version and you have the new utilities kit tar file, you do not need to use the emlxu_remove script; the emlxu_install script removes any old version as it installs the newer version; see “Installing or Updating the Utilities Using the emlxu_install Script” on page 1 for more details.

If you do not want to update the utilities package, and only want to uninstall it, use the emlxu_remove script.

To remove all emlxu files:

1. Log in as root, or su to root.
2. Go to the directory where the emlxu_remove script is located, or to the directory where the original utilities kit tar file is located by typing:

   ```
   cd <directory>
   ```

3. If you have the emlxu_remove script, skip to step 4. If you do not have the emlxu_remove script but you do have the original emlxu utilities kit tar file, extract the emlxu_remove script from the tar file by typing:

   ```
   tar xf emlxu_kit-<version>-sparc.tar emlxu_remove
   ```

4. Remove the emlxu utilities package by typing:

   ```
   emlxu_remove
   ```

5. The script locates the EMLXemlxu utilities package, and the following message is displayed:

   ```
   <Removing EMLXemlxu package>
   ```

   If no package is installed, a message indicates this; skip to step 7. Otherwise, you are prompted to remove the package with the following message:

   ```
   Do you want to remove this package? [y,n,?,q]
   ```

   Enter **<y>**. The following message is displayed:

   ```
   Removal of <EMLXemlxu> was successful.
   ```

6. The script performs some cleanup and the following messages are displayed:

   ```
   <Removing emlxu scripts>
   <emlxu_remove complete>
   ```

The utilities package has been removed. If you want to install another version of the emlxu utilities package, follow the instructions provided in one of the following sections:

“Installing or Updating the Utilities Using the emlxu_install Script” on page 1.

“Installing the Utilities Package Manually” on page 4.

For additional information on installing and removing packages, refer the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Installing the Utilities Package Manually

If an earlier version of the emlxu utilities package is already installed on the system and you want to install a different version, follow the instructions in “Removing the Utilities Package Manually” on page 4, then return to this section to install the new utilities package.

To install the emlxu utilities package manually:

1. Log in as root, or su to root.
2. Copy the utilities kit from your distribution medium into a directory, referred to here as <*directory*>. The utilities kit is a .tar file named something similar to:

   ```
   emlxu_kit-<version>-sparc.tar.
   ```
3. Change to the directory where the kit tar file is located by typing:

   ```
   cd <directory>
   ```
4. Extract the installation images from the tar file by typing:

   ```
   tar xvf emlxu_kit-<version>-sparc.tar
   ```
5. Install the EMLXemlxu utilities package by typing:

   ```
   pkgadd -d . EMLXemlxu
   ```
6. The package is prepared for installation, and you are prompted to confirm the installation with the following message:

   ```
   Do you want to continue with the installation of <EMLXemlxu> [y,n,?]
   ```
7. Enter **<y>**. The installation progress is indicated.
8. Examine the output for any errors or warnings. If the installation is successful, the following message is displayed near the end of the process:

   ```
   Installation of <EMLXemlxu> was successful.
   ```

The emlxu utilities installation is complete. The utility package's programs are located in the /opt/EMLXemlxu/bin directory.

You do not have to reboot the system to run a utility program, but you must either enter the program’s full path name or add the package's bin directory (/opt/EMLXemlxu/bin) to the system environment’s search path. To use the man pages provided by the package, you must also add the package's man directory (opt/EMLXemlxu/man) to the system environment's man path.

Removing the Utilities Package Manually

To manually remove the EMLXemlxu utilities package:

1. Type:

   ```
   pkgrm EMLXemlxu
   ```
2. You are prompted to confirm the removal by the following message:

   ```
   Do you want to remove this package? [y,n,?,q]
   ```
3. Enter **<y>**. The package is prepared for removal, and you are prompted again for confirmation:

   ```
   Do you want to remove this package? [y,n,?,q]
   ```
4. Enter **<y>**. The following message is displayed:

   ```
   Removal of <EMLXemlxu> was successful
   ```

For additional information on installing and removing packages, refer to the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Updating the Utilities Package Manually

To manually update the EMLXemlxu utilities package:

1. Remove the existing emlxu utilities package as detailed in “Removing the Utilities Package Manually” on page 4.
2. Install the new emlxu utilities package as detailed in “Installing the Utilities Package Manually” on page 4.

Using the emlxadm Utility

The emlxadm utility is used to change driver parameters through a local interactive or command line interface. It can also be used for firmware updates on non-Oracle branded devices. The OneCommand Manager application provides the same functions as emlxadm, and a number of additional ones, on multiple systems, through a choice of a graphical user interface and a scriptable command line interface. It is intended to be a direct user interface to the Fibre Channel input/output (FCIO) interface provided by the Oracle StorEdge SFS. The FCIO interface provides a Oracle common ioctl interface to the FCTL driver, which manages the FCA drivers for each Fibre Channel adapter attached to the host system.

Note: The OneCommand Manager application is supported only for LightPulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Modes of Operation (emlxadm)

The emlxadm utility program can be run in two modes:

- Interactive
- Command line interface (CLI)

Interactive Mode (emlxadm)

The emlxadm utility program can be run in an interactive command mode by typing the name of the program without any command line arguments. For example:

```
# emlxadm
```

After the utility is started, the emlxadm program scans the host system and prepares a list of qualified adapter ports. Qualified adapter ports are devices that attach to the emlxs driver or the oce driver. After the list is prepared, the utility presents the following information:

```
Available Emulex HBA's:

1. SFS:emlxs0   : /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0,2/fp@0,0 (CONNECTED)
2. NIC:oce0     : /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0           (CONNECTED)
3. SFS:emlxs1   : /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0,3/fp@0,0 (CONNECTED)
4. NIC:oce1     : /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0,1         (CONNECTED)
5. FCT:emlxs2   : /devices/pci@0,0/pci10de,5d@d/pci10df,f100@0           (CONNECTED)
6. SFS:emlxs3   : /devices/pci@0,0/pci10de,5d@d/pci10df,f100@0,1/fp@0,0 (CONNECTED)

Enter an HBA number or zero to exit:
```

Enter an HBA number or zero to exit: You must choose from one of the available adapters in the list by entering the appropriate number. In this example, the term SFS indicates that the corresponding adapter port is attached to the Oracle SFS stack. The term FCT indicates that the corresponding adapter port is attached to the Oracle COMSTAR stack. The term NIC indicates that the corresponding adapter port is attached to the Oracle networking stack. If you select the adapter port that has attached type SFS or

FCT, the utility presents the following list of command options:

```
HBA 1: /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0,2/fp@0,0

Available commands:        [FCIO rev2]

get_num_devs                - Returns the number of FC devices seen by this HBA.
get_dev_list                - Returns a list of FC devices seen by this HBA.
get_logi_params <wwpn>      - Returns the login paramters for a specified FC device.
get_host_params             - Return the host parameters.
get_sym_pname               - Returns the symbolic port name of a device.
set_sym_pname <string>      - Sets the symbolic port name for a device.
get_sym_nname               - Returns the symbolic node name of a device.
set_sym_nname <string>      - Sets the symbolic node name for a device.
dev_login <wwpn>            - Performs an FC login to a device.
dev_logout <wwpn>           - Performs an FC logout to a device.
get_state <wwpn>            - Returns current SFS state of a specified device.
dev_remove <wwpn>           - Remove the FC device from SFS management.
link_status <d_id>          - Request link error status from a specified D_ID.
get_fcode_rev               - Returns the current Fcode revison of the HBA.
download_fcode [filename]   - Download the HBA fcode.
get_fw_rev                  - Returns the current firmware revison of the HBA.
download_fw [filename]      - Download the HBA firmware.
get_boot_rev                - Returns the current boot revison of the HBA.
download_boot [filename]    - Download the HBA boot image.
get_dump_size               - Returns the HBA's firmware core dump size.
force_dump                  - Force a firmware core dump on this HBA.
get_dump [-h]               - Saves firmware core dump to a file.
get_topology                - Returns the current FC network topology.
reset_link [wwpn]           - Resets the link of a specified public loop FC device.
reset_hard                  - Reset the HBA.
diag ...                    - Perform a diagnostic test on the HBA.
ns                          - Performs a complete query of the fabric name server.
parm_get_num                - Returns the total number of configurable parameters.
parm_get_list               - Returns a list of configurable parameters.
parm_get <label>            - Gets the value of a specified parameter in the driver.
parm_set <label> <val>      - Sets the value of a specified parameter in the driver.
msgbuf [all], <number> [-i interval] - Returns the driver's internal message log.
get_host_attrs              - Returns the host adapter and port attributes.
get_port_attrs <index>, <wwn>, all   - Returns the port attributes.
get_path <index>            - Returns the adapter path.
get_vpd                     - Returns the adapter's Vital Product Data (VPD).
boot_code [enable, disable] - Sets or shows the boot code state in this HBA.
get_rnid [wwpn]             - Gets the RNID information for local or specified port.
get_inst                    - Gets the driver instance.
get_phy_attrs- Returns the current PHY attributes for the HBA. (CNA's only)
q                           - Exits this program.
h                           - Returns this help screen.
hba                         - Select another hba.
p                           - Repeat previous command.

emlxadm>
```

At the bottom of the command list is an emlxadm> prompt. From this point, the utility is prompt driven. When the prompt is displayed, you must enter one of the commands in the list. The list is displayed automatically only once, but you can display it again by typing **<h>** at the prompt. To exit the program, type **<q>**.

Some commands require additional arguments, such as a FC World Wide Port Name (WWPN) or a FC port address (D_ID). To display the available arguments for a command, enter the command without any arguments.

For example, the command get_state requires a WWPN for the target device. If only the command without the argument is entered, the following statement appears to indicate that the command requires an argument to be executed. For example:

```
emlxadm> get_state
Usage: get_state <wwpn>
emlxadm> get_state 21000020371938fa
State: PORT_DEVICE_LOGGED_IN
```

If you select the adapter port that has the attachment type NIC, the utility presents the following list of command options:

```
HBA 2: /devices/pci@0,0/pci10de,5d@c/pci10df,e602@0 (physical port)

Available commands:      [NIC rev1]

get_hba_attrs           - Returns the current control attributes for the HBA.
get_linkinfo            - Returns the current link status information for the HBA port.
get_fw_rev              - Returns the current firmware revision of the HBA.
download_fw [filename]  - Download the HBA firmware.
q                       - Exits this program.
h                       - Returns this help screen.
hba                     - Select another HBA.
p                       - Repeat previous command.

emlxadm>
```

CLI Mode (emlxadm)

There are two ways to run emlxadm in a CLI mode:

- Device path option
- Instance option

Device Path Option

```
USAGE: emlxadm <device path> [-y] <cmd> [cmd_options]
```

Run the emlxadm utility program using the device path option by typing the name of the program, followed by the full device name of the desired adapter (or a pattern string for multiple adapters), followed by a valid command and any required command arguments. In the following example, the emlxadm utility pauses to ask if you want to continue before executing the command. To specify a full device name, type:

```
# emlxadm /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl get_state 21000020371938fa
```

Information similar to the following is displayed:

```
Found path to 1 HBA port(s).
HBA port:  /devices/pci@1e,6000000/SUNW,emlxs@2/fp@0,0:devctl
     >Do you wish to continue with this device [y,n,q] ? y <---Response required

State: PORT_DEVICE_LOGGED_IN
#
```

If you do not want the utility to pause for verification, add a "-y" option just after the device path, and the emlxadm utility will skip the verification. For example:

```
# emlxadm /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl  -y  get_state
21000020371938fa
```

Information similar to the following is displayed:

```
Found path to 1 HBA port(s).
HBA port:  /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl
State: PORT_DEVICE_LOGGED_IN

#
```

If you want to run a command on multiple adapters, you can use a pattern string instead of a full device path. If the entire pattern string matches any part of an adapter device path, the command executes against that adapter. Again, in this example the emlxadm utility pauses to ask if you want to continue before executing the command.

For example:

```
# emlxadm  "SUNW,emlxs@2"  get_num_devs
```

Information similar to the following is displayed:

```
Found path to 2 HBA port(s).
HBA port:  /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl
        > Do you wish to continue with this device [y,n,q] ?  y   <-- (Response required)

There are 5 devices reported on this port.

HBA port:  /devices/pci@1e,600000/SUNW,emlxs@2,1/fp@0,0:devctl
        > Do you wish to continue with this device [y,n,q] ?  y   <-- (Response required)
There are 0 devices reported on this port.
#
```

If you do not want the utility to pause for verification, add a "-y" option just after the pattern string, and the emlxadm utility will skip the verification. For example:

```
# emlxadm  "SUNW,emlxs@2"  -y  get_num_devs
```

Information similar to the following is displayed:

```
Found path to 2 HBA port(s).
HBA port:  /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl

There are 5 devices reported on this port.

HBA port:  /devices/pci@1e,600000/SUNW,emlxs@2,1/fp@0,0:devctl

There are 0 devices reported on this port.
#
```

Instance option

```
USAGE:   emlxadm -i<N, SFS, FCT, or all> [-y] <cmd> [cmd_option(s)]
```

Run the emlxadm utility program using the instance option by typing the name of the program, followed by one of the options listed below, followed by a valid command and any required command arguments.

```
 <N>  - A specific emlxs driver instance. (N=1 means emlxs1, N=2 means emlxs2).
SFS   - All emlxs driver instances attached to the Oracle SFS interface.
FCT   - All emlxs driver instances attached to the Oracle COMSTAR interface.
all   - All emlxs driver instances.
```

This mode of operation enables you to use the emlxadm utility as part of a script or another program capable of executing system level calls.

Using Help

The emlxadm utility offers online help for the commands that you use. Type <*emlxadm help*> at the prompt and the CLI usage screen opens. For example

```
USAGE:   emlxadm
         or                    :Runs utility in interactive mode.

         emlxadm -v

         or

         emlxadm -i<N, SFS, FCT, or all> [-y] <cmd> [cmd_option(s)]

         or

         emlxadm -j<N, SFS, FCT, or all> [-y] <cmd> [cmd_option(s)]

         or

         emlxadm -n<N or all> [-y] <cmd> [cmd_option(s)]

         or

         emlxadm -m<N or all> [-y] <cmd> [cmd_option(s)]

         or

         emlxadm <device_path> [-y] <cmd> [cmd_option(s)]

OPTIONS:

         -v           Display utility version information.

         -i<N>        Executes command on a specific emlxs driver instance.
                      (Example: N=2 for emlxs2 or N=2.1 for emlxs2.1)

         -iSFS        Executes command on all SFS emlxs driver instances.

         -iFCT        Executes command on all FCT emlxs driver instances.
```

-iall	Executes command on all emlxs driver instances.
-j<N>	Executes command on a specific elxfc driver instance. (Example: N=2 for elxfc2 or N=2.1 for elxfc2.1)
-jSFS	Executes command on all SFS elxfc driver instances.
-jFCT:	Executes command on all FCT elxfc driver instances.
-jall:	Executes command on all elxfc driver instances.
-n<N>:	Executes command on a specific oce driver instance. (Example: N=2 for oce2 or N=5 for oce5)
-nall:	Executes command on all oce driver instances.
-m<N>:	Executes command on a specific elxnic driver instance. (Example: N=2 for elxnic2 or N=5 for elxnic5)
-mall:	Executes command on all oce driver instances.
device_path:	If a full device path is not specified, then the command will be executed on all device paths containing the specified device_path string.
-y:	If multiple devices are found, the utility will ask for verification before executing the command on each device. This option will cause the utility to skip the verification and automatically execute the command on each device.

Command Descriptions (emlxadm)

This section provides a list of commands and descriptions that can be issued with the emlxadm utility.

Table 1: Alphabetical Listing of All Commands

Command	Description	For an example, refer to:
boot_code [enable or disable]	Sets or shows the boot code state of the current adapter.	page 14
dev_login <wwpn>	Performs an FC login to an FC device on the network, if not already logged in.	page 14
dev_logout <wwpn>	Performs an FC logout to an FC device on the network, if not already logged out.	page 14
dev_remove <wwpn>	Removes the specified FC device from Leadville management. **Warning:** This command is currently not properly supported in the Leadville stack and will cause the host operating system to panic.	page 15
diag <test [parameters]> or diag code <cmd_code (hex)>	Performs the specified diagnostics function or command code on the adapter port. This command provides support for the Emulex-specific tests shown below, or generic support to issue an adapter-specific diagnostic code (in hexadecimal) to any third party adapter.	page 15

Table 1: Alphabetical Listing of All Commands (Continued)

Command	Description	For an example, refer to:
download_boot <*filename*>	Downloads the specified boot image file to the adapter.	page 15
download_fcode <*filename*>	Downloads the specified FCode image file to the adapter.	page 16
download_fw <*filename*>	Downloads the specified firmware image file to the adapter.	page 16
force_dump	Force a firmware core dump on the adapter.	page 16
get_boot_rev	Returns the current boot revision of the adapter.	page 17
get_dev_list	Returns a list of FC devices currently seen by this adapter port.	page 17
get_dump_size	Returns the adapter's firmware core dump size.	page 18
get_dump [-h]	Saves firmware core dump to a file.	page 18
get_fcode_rev	Returns the current FCode revision of the adapter.	page 18
get_fw_rev	Returns the current firmware revision of the adapter.	page 18
get_host_attrs	Displays all of the current host adapter API attributes.	page 18
get_host_params	Returns the FC login parameters of this adapter port.	page 20
get_inst	Returns the driver instance.	page 21
get_logi_params <*wwpn*>	Returns the FC login common service parameters for a specified FC device on the network.	page 21
get_num_devs	Returns the number of FC devices currently seen by this adapter port.	page 16
get_path <*index*>	Shows the current Solaris device path for a specified adapter port. The total number of ports available can be seen in the "Number of HBA ports" attribute displayed using the get_host_attrs command. The <index> argument is an index into this list.	page 22
get_phy_attrs	Shows the current PHY (physical transceiver connection) attributes of the adapter. This command applies to converged network adapters only.	page 22
get_port_attrs <*index*>, <*wwn*> or all	Shows the current adapter API port attributes. All of the port attributes can be displayed, or a single port can be specified by <index> or <wwn>. The total number of ports available can be seen in the "Number of Discovered Ports" attribute displayed using the get_host_attrs command. The <index> argument is an index into this list.	page 22
get_rnid [wwpn]	Retrieves the RNID information for the local or specified port.	page 24

Table 1: Alphabetical Listing of All Commands (Continued)

Command	Description	For an example, refer to:
get_state <*wwpn*>	Returns the current Leadville state of the specified FC device on the network.	page 25
get_sym_nname	Returns the symbolic FC node name of the adapter port. **Note:** This operation is currently not supported by the Solaris Leadville stack.	page 25
get_sym_pname	Returns the symbolic FC port name of the adapter port. **Note:** This operation is currently not supported by the Solaris Leadville stack.	page 25
get_topology	Returns the FC network topology of the adapter port.	page 25
get_vpd	Shows the current adapter's vital product data (VPD).	page 25
h	Shows a help menu of utility commands.	page 26
hba	Allows you to select another adapter with which to interface. This prevents you from having to exit and reenter the program.	page 26
link_status <*d_id*>	Requests and returns the current link error status from the FC device specified by the D_ID address.	page 27
msgbuf all or <*number*> [-i interval]	Shows all or part (the last <number> of lines) of the current driver message log, and can update the screen every <interval> seconds if desired. To stop the program from updating the screen, press **<Ctrl>**+ **<C>**. If no interval is provided, the current message log is displayed with no additional updates, and the emlxadm prompt returns.	page 27
ns	Performs and returns a complete query of the fabric name server.	page 27
p	Repeats the last command.	page 32
parm_get <*label*>	Retrieves the value of a specified parameter in the driver.	page 29
parm_get_list	Returns a list of configurable parameters.	page 29
parm_get_num	Returns the total number of configurable parameters.	page 28
parm_set <*label*> <*value*>	Sets the value of a specified parameter in the driver. Only dynamic parameters can be set.	page 32
q	Exits the utility program.	page 32

Table 1: Alphabetical Listing of All Commands (Continued)

Command	Description	For an example, refer to:
reset_hard	Forces the adapter to perform a hardware reset.	page 33
reset_link <*wwpn or zero for local link*>	Resets the local link, if zero is specified, or the link of a specified public loop FC device on the network.	page 33
set_sym_nname <"*string*">	Sets the symbolic FC node name of the adapter to the string provided. **Note:** This operation is currently not supported by the Solaris Leadville stack.	page 33
set_sym_pname <*"string"*>	Sets the symbolic FC port name of the adapter to the string provided. **Note:** This operation is currently not supported by the Solaris Leadville stack.	page 33

Command Usage

boot_code [enable or disable]

Sets or shows the boot code state of the current adapter.

Example:

```
emlxadm> boot_code
Boot code: Disabled

emlxadm> boot_code enable
Boot code: Enabled

emlxadm> boot_code disable
Boot code: Disabled
```

dev_login <*wwpn*>

Performs an FC login to an FC device on the network, if not already logged in.

Example:

```
emlxadm> dev_login 21000020371938fa
Done.
```

dev_logout <*wwpn*>

Performs an FC logout to an FC device on the network, if not already logged out.

Example:

```
emlxadm> dev_logout 21000020371938fa
Done.
```

dev_remove <*wwpn*>

Removes the specified FC device from Leadville management.

WARNING: This command is currently not properly supported in the Leadville stack and will cause the host operating system to panic.

diag <*test [parameters]*> or diag code <*cmd_code (hex)*>

Performs the diagnostics function or command code on the adapter port. This command provides support for the Emulex-specific tests shown below, or generic support to issue an adapter-specific diagnostic code (in hexadecimal) to any third party adapter.

```
Tests:
   emlx_biu [pattern]         - Performs the Bus Interface Unit test.
   emlx_echo <did> [pattern]  - Performs the ECHO test to a specified port id.
   emlx_post                  - Performs the Power-On Self Tests.

Parameters:
pattern - 4 byte hex pattern to be used for test. (e.g. 0xA5A5A5A5)
```

Example:

```
emlxadm> diag emlx_biu
Result: EMLX_DIAG_BIU: Operation successful.
```

or

```
emlxadm> diag emlx_echo fffffc
Result: EMLX_DIAG_ECHO: Operation successful.
```

or

```
emlxadm> diag emlx_post
Result: EMLX_DIAG_POST: Operation successful.
```

Example:

```
emlxadm> diag code 0x4526
Result: CODE(0x4526): 16 (0x10)
```

Note: The return status from the adapter is displayed in decimal and hexadecimal format if the diagnostic code is valid for the adapter. No interpretation of the return status is provided.

download_boot <*filename*>

Downloads the specified boot image file to the adapter.

Example:

```
emlxadm> download_boot TD190A4.PRG

Image Components: REL type   size=143416
    DWC file:       BOOT: version=03845054, 1.90a4

Current: Boot: 1.90a3
New:     Boot: 1.90a4   143416 (0x23038) bytes

Are you sure you want to download this image? (y or n): y
Downloading...
Done.
```

download_fcode <*filename*>

Downloads the specified FCode image file to the adapter.

Example:

```
emlxadm> download_fcode LP10000DC-S.fcode
Image Components: REL type   size=33848
    DWC file:       BOOT: version=03841512, 1.50a2

Current: Fcode: 1.50a2
New:     Fcode: 1.05e   33848 (0x8438) bytes

Are you sure you want to download this image? (y or n): y

Downloading...
Result: Operation successful.
Done.
```

download_fw <*filename*>

Downloads the specified firmware image file to the adapter.

Note: The Emulex FCA driver does not allow firmware updates to Oracle-branded adapters.

Example:

```
emlxadm> download_fw LP10000DC-S.fw
Image Components: NOP type
    AWC file:       KERN: version=ff801315, 1.30a5
    DWC file:       SLI2: version=07831914, 1.90a4
    DWC prog:       TEST: version=00f51010, 1.00a0
    DWC prog:       STUB: version=02881914, 1.90a4
    DWC prog:       SLI1: version=06831914, 1.90a4
    DWC prog:       SLI2: version=07831914, 1.90a4

Current: Firmware: 1.90a4
New:     Firmware: 1.05e 366712 (0x59878) bytes

Are you sure you want to download this image? (y or n): y

Downloading...
Done.
```

force_dump

Force a firmware core dump on the adapter.

Example:

```
emlxadm> force_dump
Done.
```

get_boot_rev

Returns the current boot revision of the adapter.

Example:

```
emlxadm> get_boot_rev
Firmware revision: LP10000DC-S 1.90a3
```

get_dev_list

Returns a list of FC devices currently seen by this adapter port.

Example:

```
emlxadm> get_dev_list
----------------------------------------------
Device 0:
          Dtype: 0
FC4_type[proto]: 0x00000100, 0x00000000, 0x00000000, 0x00000000, 0x00000000,
0x00000000, 0x00000000, 0x00000000
          State: Logged_In
           D_id: 113e1
           LILP: 0
      Hard Addr: e1
           WWPN: 21000020371938fa
           WWNN: 20000020371938fa
----------------------------------------------
Device 1:
          Dtype: 0
FC4_type[proto]: 0x00000100, 0x00000000, 0x00000000, 0x00000000, 0x00000000,
0x00000000, 0x00000000, 0x00000000
          State: Logged_In
           D_id: 113e2
           LILP: 0
      Hard Addr: e2
           WWPN: 21000020371939a2
           WWNN: 20000020371939a2
----------------------------------------------
Device 2:
          Dtype: 0
FC4_type[proto]: 0x00000100, 0x00000000, 0x00000000, 0x00000000, 0x00000000,
0x00000000, 0x00000000, 0x00000000
          State: Logged_In
           D_id: 113e4
           LILP: 0
      Hard Addr: e4
           WWPN: 21000020371938a3
           WWNN: 20000020371938a3
----------------------------------------------
Device 3:
          Dtype: 0
FC4_type[proto]: 0x00000100, 0x00000000, 0x00000000, 0x00000000, 0x00000000,
0x00000000, 0x00000000, 0x00000000
          State: Logged_In
           D_id: 113e8
```

```
          LILP: 0
     Hard Addr: e8
          WWPN: 2100002037193670
          WWNN: 2000002037193670
```

get_dump_size

Returns the adapter's firmware core dump size.

Example:

If there is a valid dump in the driver memory, you will see this.

```
emlxadm> get_dump_size
Size: 6580624 (0x646990) bytes
If there is no valid dump in the driver memory, you will see this.
emlxadm> get_dump_size
Size: 0 (0x0) bytes
```

get_dump [-h]

Saves firmware core dump to a file.

Example:

If there is no valid dump in the driver memory, you will see this.

```
emlxadm> get_dump -h
No core file available.
If there is a valid dump in the driver memory, you will see this.
emlxadm> get_dump -h
   Core size: 6580624 bytes
       files: 2
    TXT file: 13728
    DMP file: 6566876
```

get_fcode_rev

Returns the current FCode revision of the adapter.

Example:

```
emlxadm> get_fcode_rev
FCODE revision: LP10000-S 1.41a3
```

get_fw_rev

Returns the current firmware revision of the adapter.

Example:

```
emlxadm> get_fw_rev
Firmware revision: LP10000DC-S 1.90a3
```

get_hba_attrs

Returns the current control attributes for the adapter.

Example:

```
emlxadm> get_hba_attrs

HBA:
```

```
     Flash ROM Version: SE HBA ATTR VER:0000.0001
          Manufacturer: Emulex Corporation
         Support Modes: TOE,NIC,VM,FCOE_INI,LRO,0
       SEEPROM Version: 2.32
         IOCTL Version: 0x00012345
         EP Fw Version: 0x00897654
          NCSI Version:
  Def Extended Timeout: 0
                 Model: OneConnect OCe10100
           Description: BladeEngine 2 10Gb FCoE Initiator
         Serial Number: 0x42543934313832373831000000000000000000000000000000
            IP Version: IPv4
      Firmware Version: 2.701.374.2
          BIOS Version: 2.701.374.2
       REDBOOT Version: 2.0.38.0
        Driver Version:
Flash Firmware Version: 2.701.374.2
   Functions Supported: 0
        Max CDB Length: 0
         ASIC Revision: A2
     Generational GUID:
        HBA Port Count: 2
  Multifunction Device: True
         iSCSI Version: 11
   Def Linkdown Timout: 0
         Physical Port: 0
 Max Domains Supported: 1
            HBA Status: Operational
           Cache Valid: True
               HBA MTU: 8342 8342 0 0 0 0 0 0
  Firmware Post Status: 0xC000
        iSCSI Features: LIST_MODE,0
         PCI Vendor ID: 0x19a2
         PCI Device ID: 0x0700
      PCI SubVendor ID: 0x10df
      PCI SubDevice ID: 0xe602
        PCI Bus Number: 0
     PCI Device Number: 0
   PCI Function Number: 0
        Interface Type: 0
     Unique Identifier: 0
           Net Filters: 8
```

get_host_attrs

Displays all of the current host adapter API attributes.

Example:

```
emlxadm> get_host_attrs
Hoxc/'
FD/st Attributes:

    Manufacturer            = Oracle Microsystems, Inc.
```

```
    Serial Number           = BG43918495
    Model                   = LP10000DC-S
    Model Description       = EMULEX LIGHTPULSE LP10000DC-S 2GB PCI-X FIBRE CHANNEL
ADAPTER
    Node WWN                = 20000000C942097E
    Node Symbolic Name      = none
    Hardware Version        = 1001206d
    Driver Version          = 1.11f.t3 (2006.04.25.11.43)
    Optional ROM Version    = 1.50a9test1
    Firmware Version        = 1.91b5
    Vendor Specific ID      = fc00
    Number of HBA ports     = 1
    Driver Name             = Emulex-S s9-64 sparc v1.11f.t3
    Last Change             = 5
    fp Instance             = e
    Node WWN                = 20000000C942097E
    Port WWN                = 10000000C942097E
    Port Fc Id              = 011700
    Port Type               = Nport
    Port State              = Online
    Port Supported COS      = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000120, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name         = none
    Port Supported Speed       = 1Gb, 2Gb
    Port Speed                 = 1Gb
    Port Max Frame Size        = 0x800 bytes
    Fabric Name                = 0000000000000000
    Number of Discovered Ports = 4
```

get_host_params

Returns the FC login parameters of this adapter port.

Example:

```
emlxadm> get_host_params

Host:
          Dtype: 0
FC4_type[proto]: 0x00000120, 0x00000000, 0x00000000, 0x00000000, 0x00000000,
0x00000000, 0x00000000, 0x00000000
          State: Online
      Linkspeed: 1Gb
           D_id: 11700
           LILP: 5
      Hard Addr: 0
           WWPN: 10000000c942097e
           WWNN: 20000000c942097e
```

get_inst

Returns driver instance for this adapter port.

Example:

```
emlxadm> get_inst
emlxs0
```

get_linkinfo

Returns the current link status information for the adapter port.

Example:

```
emlxadm> get_linkinfo

      Physical port: 0
         MAC Duplex: Full Duplex
          MAC Speed: 10 Gbps
          MAC Fault: None
    Mgmt MAC Duplex: None
     Mgmt MAC Speed: None
     QOS Link Speed: Disabled
Logical Link Status: Link Up
```

get_logi_params <*wwpn*>

Returns the FC login common service parameters for a specified FC device on the network.

Example:

```
emlxadm> get_logi_params 21000020371938fa
Login Parameters:
00 00 00 00
20 20 00 00
88 00 08 00
00 ff 00 02
00 00 01 f4
21 00 00 20
37 19 38 fa
20 00 00 20
37 19 38 fa
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
80 00 00 00
00 00 08 00
00 ff 00 00
00 01 00 00
00 00 00 00
00 00 00 00
00 00 00 00
```

```
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
```

get_num_devs

Returns the number of FC devices currently seen by this adapter port.

Example:

```
emlxadm>   get_num_devs
There are 4 devices reported on this port.
```

get_path <*index*>

Shows the current Solaris device path for a specified adapter port. The total number of ports available can be seen in the "Number of HBA ports" attribute displayed using the get_host_attrs command. The <index> argument is an index into this list.

Example:

```
emlxadm> get_path 0
Adapter: /pci@1e,600000/SUNW,emlxs@2/fp@0,0

emlxadm> get_path 1
Adapt
er: /pci@1e,600000/SUNW,emlxs@2,1/fp@0,0
```

get_phy_attrs

Shows the current PHY (physical transceiver connection) attributes of the for the HBA. This command applies to converged network adapters only.

Example:

```
   emlxadm> get_phy_attrs

PHY Attributes:
            PHY Type: XAUI
            Interface Type: CX4 10 GB
            Flags: 0x00000000
```

get_port_attrs <*index*>, <*wwn*> or all

Shows the current adapter API port attributes. All of the port attributes can be displayed, or a single port can be specified by <index> or <wwn>. The total number of ports available can be seen in the "Number of Discovered Ports" attribute displayed using the get_host_attrs command. The <index> argument is an index into this list.

Example:

```
emlxadm> get_port_attrs all
Host Port Attributes:

    Last Change               = 5
    fp Instance               = e
```

```
    Node WWN                    = 20000000C942097E
    Port WWN                    = 10000000C942097E
    Port Fc Id                  = 011700
    Port Type                   = Nport
    Port State                  = Online
    Port Supported COS          = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000120, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name          = none
    Port Supported Speed        = 1Gb, 2Gb
    Port Speed                  = 1Gb
    Port Max Frame Size         = 0x800 bytes
    Fabric Name                 = 0000000000000000
    Number of Discovered Ports = 4

Port[0] Attributes:

    Node WWN                    = 20000020371938FA
    Port WWN                    = 21000020371938FA
    Port Fc Id                  = 0113e1
    Port Type                   = Unknown
    Port State                  = Unknown
    Port Supported COS          = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name          = SEAGATE ST39103FC       0004
    Port Supported Speed        = Unknown
    Port Speed                  = Unknown
    Port Max Frame Size         = 0x0 bytes
    Fabric Name                 = 0000000000000000

Port[1] Attributes:

    Node WWN                    = 20000020371938A2
    Port WWN                    = 21000020371938A2
    Port Fc Id                  = 0113e2
    Port Type                   = Unknown
    Port State                  = Unknown
    Port Supported COS          = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name          = SEAGATE ST39103FC       0004
```

```
    Port Supported Speed       = Unknown
    Port Speed                 = Unknown
    Port Max Frame Size        = 0x0 bytes
    Fabric Name                = 0000000000000000


Port[2] Attributes:

    Node WWN                   = 20000020371939A3
    Port WWN                   = 21000020371939A3
    Port Fc Id                 = 0113e4
    Port Type                  = Unknown
    Port State                 = Unknown
    Port Supported COS         = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name         = SEAGATE ST39103FC       0004
    Port Supported Speed       = Unknown
    Port Speed                 = Unknown
    Port Max Frame Size        = 0x0 bytes
    Fabric Name                = 0000000000000000

Port[3] Attributes:

    Node WWN                   = 2000002037193670
    Port WWN                   = 2100002037193670
    Port Fc Id                 = 0113e8
    Port Type                  = Unknown
    Port State                 = Unknown
    Port Supported COS         = Class3
    Port Supported FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Active FC4 Types:
        00000000, 00000000, 00000000, 00000000,
        00000000, 00000000, 00000000, 00000000,
    Port Symbolic Name         = SEAGATE ST39103FC       0004
    Port Supported Speed       = Unknown
    Port Speed                 = Unknown
    Port Max Frame Size        = 0x0 bytes
    Fabric Name                = 0000000000000000
```

get_rnid [wwpn]

Returns the RNID information for the local or specified port.

Example:

```
emlxadm> get_rnid
Offset: 00 01 02 03  04 05 06 07   08 09 0A 0B   0C 0D 0E 0F   ASCII:
-------------------------------------------------------------------------------
```

```
 0:     31 30 30 30  30 30 30 30   63 39 34 63   62 64 34 65   10000000c94cbd4e
10:     00 00 00 07  00 00 00 00   00 00 00 00   00 01 00 00   ................
20:     00 00 00 00  00 00 00 00   00 00 00 00   00 00 00 00   ................
30:     00 00 00 00                                           ....
Done.
```

get_state <*wwpn*>

Returns the current Leadville state of the specified FC device on the network.

Example:

```
emlxadm> get_state 21000020371938fa

State: PORT_DEVICE_LOGGED_IN
```

get_sym_pname

Returns the symbolic FC port name of the adapter port.

Note: This operation is currently not supported by the Solaris Leadville stack.

Example:

```
emlxadm> get_sym_pname
ioctl: FCIO_GET_SYM_PNAME: Operation not supported
```

get_sym_nname

Returns the symbolic FC node name of the adapter port.

Note: This operation is currently not supported by the Solaris Leadville stack.

Example:

```
emlxadm> get_sym_nname
ioctl: FCIO_GET_SYM_NNAME: Operation not supported
```

get_topology

Returns the FC network topology of the adapter port.

Example:

```
emlxadm> get_topology
Topology: PRIVATE_LOOP
```

get_vpd

Displays the current adapter's vital product data (VPD).

Example:

```
emlxadm> get_vpd
Vital Product Data:
     Identifier (ID): FC2G PCI-X LP10000DC - Oracle
    Part Number (PN): LP10000DC-S
   Manufacturer (MN): Oracle.
  Serial Number (SN): BG43918495
    Description (V1): EMULEX LIGHTPULSE LP10000DC-S 2GB PCI-X FIBRE CHANNEL ADAPTER
          Model (V2): LP10000DC-S
```

```
  Program Types (V3): T2:83,88,T3:84,T6:83,T7:83,TB:83,TFF:80
    Port Number (V4): 0
```

h

Shows a help menu of utility commands.

Example:

```
emlxadm> h

Available commands:                    [FCIO rev2]

   get_num_devs -          Returns the number of FC devices seen by this HBA.
   get_dev_list -          Returns a list of FC devices seen by this HBA.
get_logi_params <wwpn> -   Returns the login parameters for a specified FC device.
get_host_params -          Return the host parameters.
  get_sym_pname -          Returns the symbolic port name of a device.
  set_sym_pname <string> - Sets the symbolic port name for a device.
  get_sym_nname -          Returns the symbolic node name of a device.
  set_sym_nname <string> - Sets the symbolic node name for a device.
      dev_login <wwpn> -   Performs an FC login to a device.
     dev_logout <wwpn> -   Performs an FC logout to a device.
      get_state <wwpn> -   Returns current SFS state of a specified device.
     dev_remove <wwpn> -   Remove the FC device from SFS management.
    link_status <d_id> -    Request link error status from a specified D_ID.
  get_fcode_rev -           Returns the current Fcode revision of the HBA.
 download_fcode [filename] - Download the HBA fcode.
     get_fw_rev -           Returns the current firmware revison of the HBA.
    download_fw [filename] - Download the HBA firmware.
   get_boot_rev -           Returns the current boot revison of the HBA.
  download_boot [filename] - Download the HBA boot image.
     reset_link [wwpn] -    Resets the link of a specified FC device.
     reset_hard -           Resets the HBA.
      diag... -             Perform a diagnostic test on the HBA.
            ns -            Performs a complete query of the fabric name server.
   parm_get_num -           Returns the total number of configurable parameters.
  parm_get_list -           Returns a list of configurable parameters.
       parm_get <label> -   Gets the value of a specified parameter in the driver.
       parm_set <label> <val> - Sets the value of a specified parameter in the driver.
         msgbuf [all], <number> [-i interval] - Returns the driver's internal message log.
 get_host_attrs -    Returns the host adapter and port attributes.
 get_port_attrs <index>, <wwn>, all - Returns the port attributes.
       get_path <index> - Returns the adapter path.
        get_vpd -    Returns the adapter's Vital Product Data (VPD).
      boot_code [enable, disable] - Sets or shows the boot code state in this HBA.
       get_rnid [wwpn] - Gets the RNID information for the local or specified port.
       get_inst -    Gets the driver instance
get_phy_attrs                      -Returns the current PHY attributes for the HBA.(CNA's
Only)
             q -    Exits this program.
             h -    Returns this help screen.
           hba -    Select another hba.
             p -    Repeat previous command.

emlxadm>
```

hba

Select another adapter with which to interface. This prevents you from having to exit and reenter the program.

Example:

```
Available Emulex HBA's:

1. SFS:emlxs0 : /devices/pci@7c0/pci@0/pci@1/pci@0,2/SUNW,emlxs@1/fp@0,0 (CONNECTED)
2. FCT:emlxs1 : /devices/pci@7c0/pci@0/pci@1/pci@0,2/SUNW,emlxs@1,1 (CONNECTED)

Enter an HBA number or zero to exit:
```

link_status <*d_id*>

Requests and returns the current link error status from the FC device specified by the d_id address.

Example:

```
emlxadm> link_status e8

D_ID: e8
            Link failures: 3 (0x3)
       Loss of sync count: 12 (0xc)
     Loss of signal count: 0 (0x0)
Primitive sequence errors: 0 (0x0)
         Invalid tx words: 17 (0x11)
        Invalid CRC count: 0 (0x0)
```

msgbuf all or <*number*> [-i interval]

Shows all or part (the last <number> of lines) of the current driver message log, and can update the screen every <interval> seconds if desired. To stop the program from updating the screen, press **<Ctrl>**+ **<C>**. If no interval is provided, the current message log is displayed with no additional updates, and the emlxadm prompt returns.

Example:

```
emlxadm> msgbuf 10

  155130.01: 1002033:[B.1C35]emlxs0:  DEBUG: 800: ELS sent.  (GA_NXT: did=fffffc
[00011000,00000000])
  155130.02: 1002034:[4.00C9]emlxs0:  DEBUG: 801: ELS comp.  (GA_NXT: CT_ACC: Rsn=0
Exp=0 [020113e1,21000020])
  155130.02: 1002035:[B.1C35]emlxs0:  DEBUG: 800: ELS sent.  (GA_NXT: did=fffffc
[000113e1,00000000])
  155130.02: 1002036:[4.00C9]emlxs0:  DEBUG: 801: ELS comp.  (GA_NXT: CT_ACC: Rsn=0
Exp=0 [020113e2,21000020])
  155130.02: 1002037:[B.1C35]emlxs0:  DEBUG: 800: ELS sent.  (GA_NXT: did=fffffc
[000113e2,00000000])
  155130.02: 1002038:[4.00C9]emlxs0:  DEBUG: 801: ELS comp.  (GA_NXT: CT_ACC: Rsn=0
Exp=0 [020113e4,21000020])
  155130.03: 1002039:[B.1C35]emlxs0:  DEBUG: 800: ELS sent.  (GA_NXT: did=fffffc
[000113e4,00000000])
  155130.03: 1002040:[4.00C9]emlxs0:  DEBUG: 801: ELS comp.  (GA_NXT: CT_ACC: Rsn=0
Exp=0 [020113e8,21000020])
  155130.03: 1002041:[B.1C35]emlxs0:  DEBUG: 800: ELS sent.  (GA_NXT: did=fffffc
[000113e8,00000000])
```

ns

Performs and returns a complete query of the fabric name server.

Example:

```
emlxadm> ns

Nameserver:
-----------------------------------------------------------------
     TYPE: Lport
      PID: 0113E1
     WWPN: 21000020371938fa
PORT_NAME: (SEAGATE ST39103FC       0004)
     WWNN: 20000020371938fa
NODE_NAME: (null)
      IPA: ffffffffffffffff
  IP_ADDR: 0.0.0.0
    CLASS: Class3
FC4_TYPES: 00000100,00000000,00000000,00000000,00000000,00000000,00000000,00000000
-----------------------------------------------------------------
     TYPE: Lport
      PID: 0113E2
     WWPN: 21000020371939a2
PORT_NAME: (SEAGATE ST39103FC       0004)
     WWNN: 20000020371939a2
NODE_NAME: (null)
       IPA: ffffffffffffffff
  IP_ADDR: 0.0.0.0
    CLASS: Class3
FC4_TYPES: 00000100,00000000,00000000,00000000,00000000,00000000,00000000,00000000
-----------------------------------------------------------------
     TYPE: Lport
      PID: 0113E4
     WWPN: 21000020371938a3
PORT_NAME: (SEAGATE ST39103FC       0004)
     WWNN: 20000020371938a3
NODE_NAME: (null)
      IPA: ffffffffffffffff
  IP_ADDR: 0.0.0.0
    CLASS: Class3
FC4_TYPES: 00000100,00000000,00000000,00000000,00000000,00000000,00000000,00000000
-----------------------------------------------------------------
     TYPE: Lport
      PID: 0113E8
     WWPN: 2100002037193670
PORT_NAME: (SEAGATE ST39103FC       0004)
     WWNN: 2000002037193670
NODE_NAME: (null)
      IPA: ffffffffffffffff
  IP_ADDR: 0.0.0.0
    CLASS: Class3
FC4_TYPES: 00000100,00000000,00000000,00000000,00000000,00000000,00000000,00000000
```

parm_get_num

Returns the total number of configurable parameters.

Example:

```
emlxadm> parm_get_num
Result: There are 18 configurable parameters in the driver.
```

parm_get *<label>*

Gets the value of a specified parameter in the driver.

Example:

```
emlxadm> parm_get adisc-support
  label: adisc-support
    min: 0
current: 1
    max: 2
default: 1
dynamic: yes
   desc: Sets the Fibre Channel ADISC login support level.
```

parm_get_list

Returns a list of configurable parameters.

Example:

```
emlxadm> parm_get_list

Parameter:
------------------------------------------------------------------
  label: console-notices
    min: 0x0
current: 0x0
    max: 0xffffffff
default: 0x0
dynamic: yes
   desc: Verbose mask for notice messages to the console.
------------------------------------------------------------------
  label: console-warnings
    min: 0x0
current: 0x0
    max: 0xffffffff
default: 0x0
dynamic: yes
   desc: Verbose mask for warning messages to the console.
------------------------------------------------------------------
  label: console-errors
    min: 0x0
current: 0x0
    max: 0xffffffff
default: 0x0
dynamic: yes
   desc: Verbose mask for error messages to the console.
------------------------------------------------------------------
  label: log-notices
    min: 0x0
current: 0xffffffff
    max: 0xffffffff
default: 0xffffffff
```

```
dynamic: yes
   desc: Verbose mask for notice messages to the messages file.
----------------------------------------------------------------
  label: log-warnings
    min: 0x0
current: 0xffffffff
    max: 0xffffffff
default: 0xffffffff
dynamic: yes
   desc: Verbose mask for warning messages to the messages file.
----------------------------------------------------------------
  label: log-errors
    min: 0x0
current: 0xffffffff
    max: 0xffffffff
default: 0xffffffff
dynamic: yes
   desc: Verbose mask for error messages to the messages file.
----------------------------------------------------------------
  label: num-iocbs
    min: 128
current: 1024
    max: 10240
default: 1024
dynamic: no
   desc: Number of outstanding IOCBs driver can queue to adapter.
----------------------------------------------------------------
  label: ub-bufs
    min: 40
current: 1000
    max: 16320
default: 1000
dynamic: no
   desc: Number of unsolicited buffers the driver should allocate.
----------------------------------------------------------------
  label: network-on
    min: 0
current: 1
    max: 1
default: 1
dynamic: no
   desc: Enable IP processing.
----------------------------------------------------------------
  label: ack0
    min: 0
current: 0
    max: 1
default: 0
dynamic: no
   desc: Enable ACK0 support.
----------------------------------------------------------------
  label: topology
    min: 0
current: 0
```

```
    max: 6
default: 0
dynamic: no
   desc: Select Fibre Channel topology.
----------------------------------------------------------------
  label: link-speed
    min: 0
current: 0
    max: 4
default: 0
dynamic: no
   desc: Select link speed.
----------------------------------------------------------------
  label: num-nodes
    min: 2
current: 512
    max: 512
default: 512
dynamic: no
   desc: Number of fibre channel nodes (NPorts) the driver will support.
----------------------------------------------------------------
  label: cr-delay
    min: 0
current: 0
    max: 63
default: 0
dynamic: no
   desc: A count of milliseconds after which an interrupt response is generated.
----------------------------------------------------------------
  label: cr-count
    min: 1
current: 1
    max: 255
default: 1
dynamic: no
   desc: A count of I/O completions after which an interrupt response is generated.
----------------------------------------------------------------
  label: assign-alpa
    min: 0x0
current: 0x0
    max: 0xef
default: 0x0
dynamic: no
   desc: Assigns a preferred ALPA to the port. Only used in Loop topology.
----------------------------------------------------------------
  label: adisc-support
    min: 0
current: 1
    max: 2
default: 1
dynamic: yes
   desc: Sets the Fibre Channel ADISC login support level.
----------------------------------------------------------------
  label: pm-support
```

```
    min: 0
current: 1
    max: 1
default: 1
dynamic: no
   desc: Enables power management support.
```

parm_set <*label*> <*value*>

Sets the value of a specified parameter in the driver. Only dynamic parameters can be set.

Example: This example sets a dynamic parameter:

```
emlxadm> parm_set adisc-support 2

  label: adisc-support
    min: 0
current: 2
    max: 2
default: 1
dynamic: yes
   desc: Sets the Fibre Channel ADISC login support level.
```

Note: To make this change permanent, you must edit the /kernel/drv/emlxs.conf file.

Example: This example attempts to set a static parameter:

```
emlxadm> parm_set network-on 1
emlxadm: EMLX_PARM_SET: Parameter (network-on) is not dynamic and cannot be changed
here.

** To make this change you must edit the /kernel/drv/emlxs.conf or **
** the /kernel/drv/emlx.conf file(s) and reboot the system.        **
```

p

Repeats the last command.

Example:

```
emlxadm> get_num_devs
There are 4 devices reported on this port.

emlxadm> p
emlxadm> get_num_devs

There are 4 devices reported on this port.
```

q

Exits the utility program.

Example:

```
emlxadm> q
Exiting...
```

reset_hard

Forces the adapter to perform a hardware reset.

Example:

```
emlxadm> reset_hard
Done.
```

reset_link <*wwpn or zero for local link*>

Resets the local link, if zero is specified, or the link of a specified FC device on the network.

Example:

```
emlxadm> reset_link 0
Done.
```

or

```
emlxadm> reset_link 21000020371938fa
Done.
```

set_sym_nname <*"string"*>

Sets the symbolic FC node name of the adapter to the string provided.

Note: This operation is currently not supported by the Solaris Leadville stack.

Example:

```
emlxadm> set_sym_nname "Emulex Corporation"
ioctl: FCIO_SET_SYM_NNAME: Operation not supported
```

set_sym_pname <*"string"*>

Sets the symbolic FC port name of the adapter to the string provided.

Note: This operation is currently not supported by the Solaris Leadville stack.

Example:

```
emlxadm> set_sym_pname "Emulex Corporation"
ioctl: FCIO_SET_SYM_PNAME: Operation not supported
```

Using the emlxdrv Utility

Use the emlxdrv utility to bind (associate) the following drivers to the various models of Fibre Channel adapters:

- emlxs - Oracle distributed Leadville Fibre Channel driver
- elxfc - Emulex distributed Leadville Fibre Channel driver
- lpfc - Emulex distributed non-Leadville Fibre Channel driver
- oce - Oracle distributed NIC driver
- elxnic - Emulex distributed NIC driver

This utility is used for managing the driver and adapter associations on the host. The emlxs and elxfc drivers can coexist on the same host. However, a specific adapter model can only be associated to one driver at a time. The same relationship also applies between the oce and elxnic drivers. Only one NIC adapter model can be associated to one NIC driver at a time.

Note: The simultaneous use of the lpfc driver and either of the emlxs or elxfc drivers is not supported and should only be used as a step when migrating complex configurations between drivers.

If you change the driver binding configuration, the host system must be rebooted in order for the new configuration to take effect.

Modes of Operation (emlxdrv)

The emlxdrv utility program can be run in two modes:

- Interactive
- CLI

Interactive Mode (emlxdrv)

Run the emlxdrv utility program in interactive mode by typing the name of the program without any command line arguments:

```
# emlxdrv
```

The emlxdrv program scans the host system and prepares a driver configuration table consisting of bindings (associations) between the emlxs, elxfc, lpfc, oce and elxnic NIC drivers and a list of Emulex Fibre Channel adapter models. After the table is prepared, the utility displays the following:

FC Driver	Alias	Pres	Boot	Oracle	emlxs	elxfc	lpfc	Models
elxfc	lpfs	-	-	-	yes	yes	yes	LP8000S, LP9002S (SBUS)
elxfc	f800	-	-	-	yes	yes	yes	LP8000, LP8000DC
elxfc	f900	-	-	-	yes	yes	yes	LP9002, LP9002C, LP9002DC, LP9402DC
elxfc	f980	-	-	-	yes	yes	yes	LP9802,LP9802DC
elxfc	fa00	-	-	-	yes	yes	yes	LP10000, LP10000DC, LP10000ExDC
elxfc	fd00	-	-	-	yes	yes	yes	LP11000, LP11002
elxfc	fe00	yes	-	-	yes	yes	yes	LPe11000, LPe11002, LPe11004

```
FC
Driver  Alias    Pres   Boot   Oracle   emlxs   elxfc   lpfc   Models
elxfc   fe12     -      -      -        yes     yes     yes    LPe11000, LPe11002,
                                                              LPe11004
elxfc   f100     -      -      -        yes     yes     yes    LPe12000, LPe12002
elxfc   f111     -      -      -        yes     yes     yes    LPe12000, LPe12002
elxfc   f112     -      -      -        yes     yes     yes    LPe12000, LPe12002
elxfc   fe05     -      -      -        yes     -yes    yes    LP21000, LP21002
elxfc   f0a5     -      -      -        yes     yes     -      2G Blade Adapter
emlxs   fc00     -      -      yes      -yes    -       -      LP10000-S, LP10000DC-S
emlxs   fc10     yes    -      yes      yes     -       -      LP11000-S, LP11002-S
emlxs   fc20     -      -      yes      yes     -       -      LPe11000-S, LPe11002-S
emxls   fc40     -      -      yes      yes     -       -      LPe12000-S, LPe12002-S
elxfc   704      -      -      -        yes     -yes    -      OCe10101, OCe10102
elxfc   714      yes    -      -        yes     yes     -      OCe11101, OCe11102

NIC
Driver  Alias    Pres   Oracle   oce     elxfc   Models
oce     700      -      -        yes     yes     OCe10101, OCe10102
elxnic  710      -      -        yes     yes     OCe11101, OCe11102

Available commands

set...     - Sets driver bindings to specified devices.
clear...   - Clears driver bindings to specified devices.
refresh... - Refresh driver bindings to current devices.
q          - Exits this program.
```

The display comprises three parts: the current driver configuration table, a list of available commands and the emlxdrv prompt (not shown above).

The driver configuration table contains the following columns of data:

- Driver - Which driver (emlxs, lpfc or "-" if none) is currently configured to bind or attach to a specific adapter alias.
- Alias - The specific adapter alias associated with a set of Emulex adapter models. Driver bindings can be made only with a specific adapter alias and not with a specific adapter model.
- Present - Whether this specific type of adapter is currently present in the host system. emlxdrv allows you to bind a driver to adapters that are not currently present in the system but that may be present at some point in the future.
- Boot - Whether this specific type of adapter is currently providing connectivity to the system's boot disk. This is important because emlxdrv does not allow you to change the driver binding to an adapter currently providing connectivity to the boot disk. If the driver binding needs to be changed to a boot device, the system must first be configured to boot through an adapter of another type. This procedure is not in the scope of this document.
- Oracle - Whether this specific type of adapter is branded and sold directly by Oracle.
- emlxs - Whether the emlxs driver supports this specific type of adapter.
- elxfc - Whether the elxfc driver supports this specific type of adapter.
- lpfc - Whether the lpfc driver supports this specific type of adapter.
- Models - A list of Emulex adapter models that are identified by a common adapter alias. Driver bindings can be made only with a specific adapter alias and not with a specific adapter model.

After the driver configuration table is a list of available commands. For a detailed explanation of each

command and its arguments, see “Command Descriptions (emlxdrv)” on page 37.

Below the command list is an **emlxdrv>** prompt. From this point, the utility is prompt driven. When the prompt is displayed, you must enter one of the commands in the list. The current driver configuration table and the available command list are displayed automatically after each command is issued.

Some commands require a <driver> or <driver_name> and an <alias> argument. You must specify one of the valid adapter aliases listed in the current driver configuration table. Each alias is shared by multiple adapter models. Driver bindings can be made only with an adapter alias and not with a specific adapter model.

CLI Mode (emlxdrv)

The emlxdrv utility program can be run in CLI mode by typing the name of the program followed by a valid command and any required command arguments. For example, you can update the a device binding by entering all the information on one line at the operating system prompt as shown below.

This mode of operation enables you to use the emlxdrv utility as part of a script or another program capable of executing system-level calls.

FC Driver	Alias	Pres	Boot	Oracle	emlxs	elxfc	lpfc	Models
elxfc	lpfs	-	-	-	yes	yes	yes	LP8000S, LP9002S (SBUS)
elxfc	f800	-	-	-	yes	yes	yes	LP8000, LP8000DC
elxfc	f900	-	-	-	yes	yes	yes	LP9002, LP9002C, LP9002DC, LP9402DC
elxfc	f980	-	-	-	yes	yes	yes	LP9802,LP9802DC
elxfc	fa00	-	-	-	yes	yes	yes	LP10000, LP10000DC, LP10000ExDC
elxfc	fd00	-	-	-	yes	yes	yes	LP11000, LP11002
elxfc	fe00	yes	-	-	yes	yes	yes	LPe11000, LPe11002, LPe11004
elxfc	fe12	-	-	-	yes	yes	yes	LPe11000, LPe11002, LPe11004
elxfc	f100	-	-	-	yes	yes	yes	LPe12000, LPe12002
elxfc	f111	-	-	-	yes	yes	yes	LPe12000, LPe12002
elxfc	f112	-	-	-	yes	yes	yes	LPe12000, LPe12002
elxfc	fe05	-	-	-	yes	-yes	yes	LP21000, LP21002
elxfc	f0a5	-	-	-	yes	yes	-	2G Blade Adapter
emlxs	fc00	-	-	yes	-yes	-	-	LP10000-S, LP10000DC-S
emlxs	fc10	yes	-	yes	yes	-	-	LP11000-S, LP11002-S
emlxs	fc20	-	-	yes	yes	-	-	LPe11000-S, LPe11002-S
emxls	fc40	-	-	yes	yes	-	-	LPe12000-S, LPe12002-S
elxfc	704	-	-	-	yes	-yes	-	OCe10101, OCe10102
elxfc	714	yes	-	-	yes	yes	-	OCe11101, OCe11102
elxfc	e200	-	-	-	yes	yes	-	OCe16001, OCe16002
elxfc	e260	-	-	-	yes	yes	-	OCe50102

NIC Driver	Alias	Pres	Oracle	oce	elxnic	Models
oce	700	-	-	yes	yes	OCe10101, OCe10102
elxnic	710	-	-	yes	yes	OCe11101, OCe11102

Command Descriptions (emlxdrv)

This section provides a list of commands that can be issued with the emlxdrv utility program.

You can view the list of commands at any time by running the emlxdrv utility in interactive mode (see “Interactive Mode (emlxdrv)” on page 34).

Table 2: Alphabetical Listing of All Commands

Command Class	Command	Description	See
Clear [driver] [device] Note: [driver] can be one of the following: all-All drivers (default) fc-All Fibre Channel drivers nic-All NIC drivers [driver_name] - Specific driver name Note: [device] can be one of the following: all - All devices (default) [alias] - Specific device alias. [driver] is ignored in this case. emulex - All Emulex-branded adapters. oracle - All Oracle-branded adapters You can combine any of the arguments from the driver list and the device list.	clear [driver] all	Clears the driver bindings to all devices. This is the default setting.	page 38
	clear [driver] [alias]	Clears the driver binding to the specified devices. The driver is ignored.	page 38
	clear [driver] emulex	Clears Emulex-branded driver bindings for a specified driver type.	page 39
	clear [driver] oracle	Clears Oracle-branded driver bindings for a specified driver type.	page 40
q	q	Exits the program.	page 40
refresh	refresh	Refreshes the host bindings and main display with the latest information.	page 40
Set [driver_name] [device] Note: [driver_name] is the name of the specific driver you wish to effect with the set comamnd. You can combine arguments from the driver list and the device list.	set [driver_name] [alias]	Sets the driver to bind to the specified devices.	page 41
	set [driver_name] all	Sets the driver to bind to all devices. This is the default setting.	page 41
	set [driver_name] oracle	Sets the driver to bind to all Oracle devices.	page 42
	set [driver_name]> emulex>	Sets the driver to bind to Emulex branded devices.	page 42

Command Usage

clear [driver] all

Clears the driver bindings to all devices based on the specified driver parameter. Select one of the following driver parameters:

- all - clears all bindings to all devices for all drivers. This is the default value for this parameter.
- fc - clears all bindings to all devices for Fibre Channel drivers.
- nic - clears all bindings to all devices for NIC drivers.
- [driver_name] - clears all bindings to all devices for a specific driver.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot. Example:

```
emlxadm>   clear all

Updating lpfs ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating f800 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating f900 ...
Cannot unload module: lpfc
Will be unloaded upon reboot.

Updating f980 ...
Cannot unload module: lpfc
Will be unloaded upon reboot.

Updating fa00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fd00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fe00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Done.
```

clear [driver] [alias]

Clears the driver bindings to specified devices based on the specified driver parameter. Select one of the following driver parameters:

- all - clears bindings to specified devices for all drivers. This is the default value for this parameter.

- fc - clears bindings to specified devices for Fibre Channel drivers.
- nic - clears bindings to specified devices for NIC drivers.
- [driver_name] - clears bindings to specified devices for a specific driver.

You must specify one of the adapter aliases listed on the screen. Each alias is shared by multiple adapter models. Driver bindings can be made only with an adapter alias and not with a specific adapter model.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv> clear dev fe00

Updating fe00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.
Done.
```

clear [driver] emulex

Clears the driver bindings to Emulex branded devices based on the specified driver parameter. Select one of the following driver parameters:

- all - clears bindings to Emulex branded devices for all drivers. This is the default value for this parameter.
- fc - clears bindings to Emulex branded devices for Fibre Channel drivers.
- nic - clears bindings to Emulex branded devices for NIC drivers.
- [driver_name] - clears bindings to Emulex branded devices for a specific driver.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv>   clear emlxs

Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fc00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fc10 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fc20 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Done.
```

clear [driver] oracle

Clears the driver bindings to all Oracle devices based on the driver parameter you specify. Select one of the following driver parameters:

- all - clears bindings to Oracle branded devices for all drivers. This is the default value for this parameter.
- fc - clears bindings to Oracle branded devices for Fibre Channel drivers.
- nic - clears bindings to Oracle branded devices for NIC drivers.
- [driver_name] - clears bindings to Oracle branded devices for a specific driver.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv> clear oracle

Updating fc00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fc10 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Updating fc20 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.

Done.
```

q

Exits the program. If changes were made to the driver bindings, a system reboot is usually required for all the changes to take effect.

Example:

```
emlxdrv> q

Exiting...

NOTE: If changes were made, then a system reboot may be required.
#
```

refresh

Refreshes the host bindings and main display with the latest information.

Example:

FC Driver	Alias	Pres	Boot	Oracle	emlxs	elxfc	lpfc Models
elxfc	lpfs	-	-	yes	yes	yes	LP8000S, LP9002S (SBUS)
elxfc	f800	-	-	yes	yes	yes	LP8000, LP8000DC
elxfc	f900	-	-	yes	yes	yes	LP9002, LP9002C, LP9002DC, LP9402DC
elxfc	f980	-	-	yes	yes	yes	LP9802DC

```
FC
Driver  Alias    Pres   Boot   Oracle   emlxs   elxfc  lpfc Models
elxfc   fa00     -      -      yes      yes     yes    LP10000, LP10000DC,
                                                       LP10000ExDC
elxfc   fd00     -      -      yes      yes     yes    LP11000, LP11002
elxfc   fe00     -      -      yes      yes     yes    LPe11000, LPe11002,
                                                       LPe11004
elxfc   f100     yes    -      yes      yes     yes    LPe12000, LPe12002
elxfc   f111     -      -      yes      yes     yes    LPe12000, LPe12002
elxfc   f112     -      -      yes      yes     yes    LPe12000, LPe12002
elxfc   fe05     -      -      yes      yes     -yes   LP21000, LP21002
elxfc   f0a5     -      -      yes      yes     -      2G Blade Adapter
-       fc00     -      yes    yes      -       -      LP10000-S, LP10000DC-S
-       fc10     -      yes    yes      -       -      LP11000-S, LP11002-S
-       fc20     yes    yes    yes      -       -      LPe11000-S, LPe11002-S
-       fc40     -      yes    yes      -       -      LPe12000-S, LPe12002-S
elxfc   704      yes    -      yes      yes     -      OCe10101, OCe10102
elxfc   714      yes    -      yes      yes     -      OCe11101, OCe11102

NIC
Driver  Alias    Pres   Oracle   oce     elxfc  Models
elxnic  700      yes    -        yes     yes    OCe10101, OCe10102
elxnic  710      yes    -        yes     yes    OCe11101, OCe11102

Available commands

set...      - Sets driver bindings to specified devices.
clear...    - Clears driver bindings to specified devices.
refresh...  - Refresh driver bindings to current devices.
q           - Exits this program.
```

set emlxs [driver_name] [alias]

Sets the driver to bind to the specified devices. You must specify one of the valid adapter aliases listed on the screen. Note that each alias is shared by multiple adapter models. Driver bindings can be made only with an adapter alias and not with a specific adapter model.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv>   set emlxs f980

Updating f980 ...
Cannot unload module: lpfc
Will be unloaded upon reboot.

Done.
```

set [driver] all

Sets the driver to bind to all devices. You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv>   set emlxs_all

Updating lpfs ...
Updating f800 ...
Updating f900 ...
Cannot unload module: lpfc
Will be unloaded upon reboot.

Updating f980 ...
Cannot unload module: lpfc
Will be unloaded upon reboot.

Updating fa00 ...
Updating fd00 ...
Updating fe00 ...
Updating fc00 ...
Updating fc10 ...
Updating fc20 ...
Done.
```

set [driver] oracle

Sets the driver to bind to all Oracle devices.

Example:

```
emlxdrv>   set emlxs_sun

Updating fc00 ...
Updating fc10 ...
Updating fc20 ...
Done.
```

set [driver name] [emulex]

Sets the driver to bind to all Emulex branded devices.

You may see the message "Cannot unload module." This indicates that you must reboot the system to get a driver to unbind from that adapter alias; emlxdrv only updates the system configuration for the next boot.

Example:

```
emlxdrv>   set lpfc fa00

Updating fa00 ...
Cannot unload module: emlxs
Will be unloaded upon reboot.
Done.
```